

# 构件化技术在学生成绩查询统计系统中的应用

李睿<sup>1</sup>, 徐红<sup>1</sup>, 曾应员<sup>2</sup>

(1. 西安建筑科技大学 土木工程学院, 陕西 西安 710055;

2 西安建筑科技大学 信息与控制工程学院, 陕西 西安 710055)

**摘要:**利用已有的软件构件来构造新的软件系统,使得用各种技术形成的软构件可以最大程度地进行重用(复用),以减少大量的重复劳动,从而大大提高生产效率。构件的可复用性越高,复用越广泛,其复用次数就会越多,价值也越大。通过对构件化技术主要思想和方法的分析,运用 PowerBuilder 所提供的典型的可复用构件 PFC,阐述了构件化技术在学生成绩查询统计系统中的应用。

**关键词:**构件化技术;PFC;可复用构件

**中图分类号:**G434;TP311

**文献标识码:**A

**文章编号:**1673-629X(2007)06-0214-03

## Application of Component Technology in Accounting System for Student Achievement

LI Rui<sup>1</sup>, XU Hong<sup>1</sup>, ZENG Ying-yuan<sup>2</sup>

(1. School of Civil Eng., Xi'an Univ. of Architecture and Techn., Xi'an 710055, China;

2 School of Info. and Control Eng., Xi'an Univ. of Architecture and Techn., Xi'an 710055, China)

**Abstract:** The technology of reconstructing new software system using available software components can make the best of reusing all kinds of technology-supported components and reduce amount of repeated endures, so that the working efficiency can be greatly improved. The stronger capability of reutilization and popularity of a component is, the higher frequency it is used and the higher its value is. By analyzing the brief theory and method of the component technology and by utilizing the typical reusing components PFC provided by PowerBuilder, applied an illustration on application of component technology in the accounting system for student achievement.

**Key words:** component technology; PFC; reusing component

## 0 引言

软件行业的工业化趋势导致了软构件的产生。能够像硬件系统那样,将部分软件组合起来构建软件系统,一直是软件行业多年来追求的目标。特别是对于很多应用软件的开发(例如很多行业和单位的 MIS 系统),若能结合系统的实际情况充分利用已有的软件构件,将会大大提高生产效率,减少大量的重复劳动。可以说,软构件技术的出现是对传统软件开发过程的一次变革。

## 1 构件的概念

构件(Component)是一种定义良好的独立、可重用

的二进制代码集合,它惟一通过接口提供服务。它是可复用的软件组成成分,可被用来构造其它软件<sup>[1]</sup>。它可以是需求分析和设计阶段的产品、代码、测试案例、文档或软件开发过程中的其他产品,如被封装的对象类、类树、一些功能模块、软件框架、软件构架(或体系结构)、文档、分析件、设计模式等。从广义上讲,软构件技术是基于面向对象的,以嵌入后马上可以使用的即插即用型软构件概念为中心,通过构件的组合来建立应用的技术体系。狭义上讲,它是通过构件组合支持应用的开发环境和系统的总称<sup>[2]</sup>。

构件技术与面向对象技术紧密相关,但也存在着很大的差异,首先在概念层面上,对象描述客观世界实体(identity),构件提供客观世界服务(service);其次在复用策略上,对象是通过继承实现复用的,而构件则将抽象的程度提到一个更高的层次,它是对一组类的组合进行封装,并代表完成一个或多个功能的特定服务,也为用户提供了多个接口。整个构件隐藏了具体的实

收稿日期:2006-08-30

基金项目:陕西省科技厅资助项目(2002F1);陕西省教育厅产业化培育项目(02JC39)

作者简介:李睿(1971-),男,内蒙古凉城人,讲师,研究方向为数据库应用技术、软件复用理论及应用。



现,只用接口提供服务。这样,在不同层次上,构件均可以将底层的多个逻辑组合成高层次上的粒度更大的新构件,甚至直接封装到一个系统,使模块的重用从代码级、对象级、架构级到系统级都可能实现,从而使软件像硬件一样,能任人装配定制而成的梦想得以实现。

生产可复用构件的目的就是能被广泛地复用,一个构件的复用次数越多,其价值也越大。为使构件能具有较高的可复用性,可复用构件应满足以下条件<sup>[3]</sup>:

(1) 构件的设计应具有较高的通用程度。

构件的可复用程度是指该构件在开发其他软件时可被复用的难易程度。构件越一般化(即通用),则其可复用程度也越高;构件越具体(即专用),则其可复用程度越低。因此为提高构件的可复用度,应尽量使构件泛化,使这些构件能在更多的待开发软件中被复用。

(2) 构件应易于调整。

虽然构件通常具有较高的通用性,然而,在特定的软件中复用该构件时都必须将构件特化,以用于具体的环境。因此,必须提供软件构件的特化机制和调整机制,使复用构件时易于调整。

(3) 构件应易于组装。

为了使构件易于组装,构件应具有良好的封装性和良好定义的接口,构件间应具有松散的耦合度,同时还应提供便于组装的机制。

(4) 构件必须具有可检索性。

构件必须具有合适的描述机制,以便开发人员能从构件库中检索到所需的构件。显然,如果构件没有很好的可检索性,那么它被复用的概率就会很低。

(5) 构件必须经过充分的测试。

构件在入库前必须经过充分的测试,尽可能多地发现并纠正构件中的缺陷,在复用过程中,当发现构件中潜藏的缺陷时,要及时更正,以使构件中的缺陷数降到最低。

## 2 学生成绩查询统计系统应用开发平台的选择

由于软构件技术是基于面向对象的,所以学生成绩查询统计系统采用了面向对象的快速应用开发(RAD)工具 PowerBuilder,它所提供的类库 PFC (Powerbuilder Foundation Class, Powerbuilder 基础类库)就是一组典型的可复用构件<sup>[4]</sup>。

## 3 学生成绩查询统计系统构件化设计与实现

### 3.1 学生成绩查询统计系统的构件化设计

学生成绩查询统计系统的构件化设计是将系统功

能划分为窗口构件、功能构件、数据接口构件以及数据构件,如图1所示。

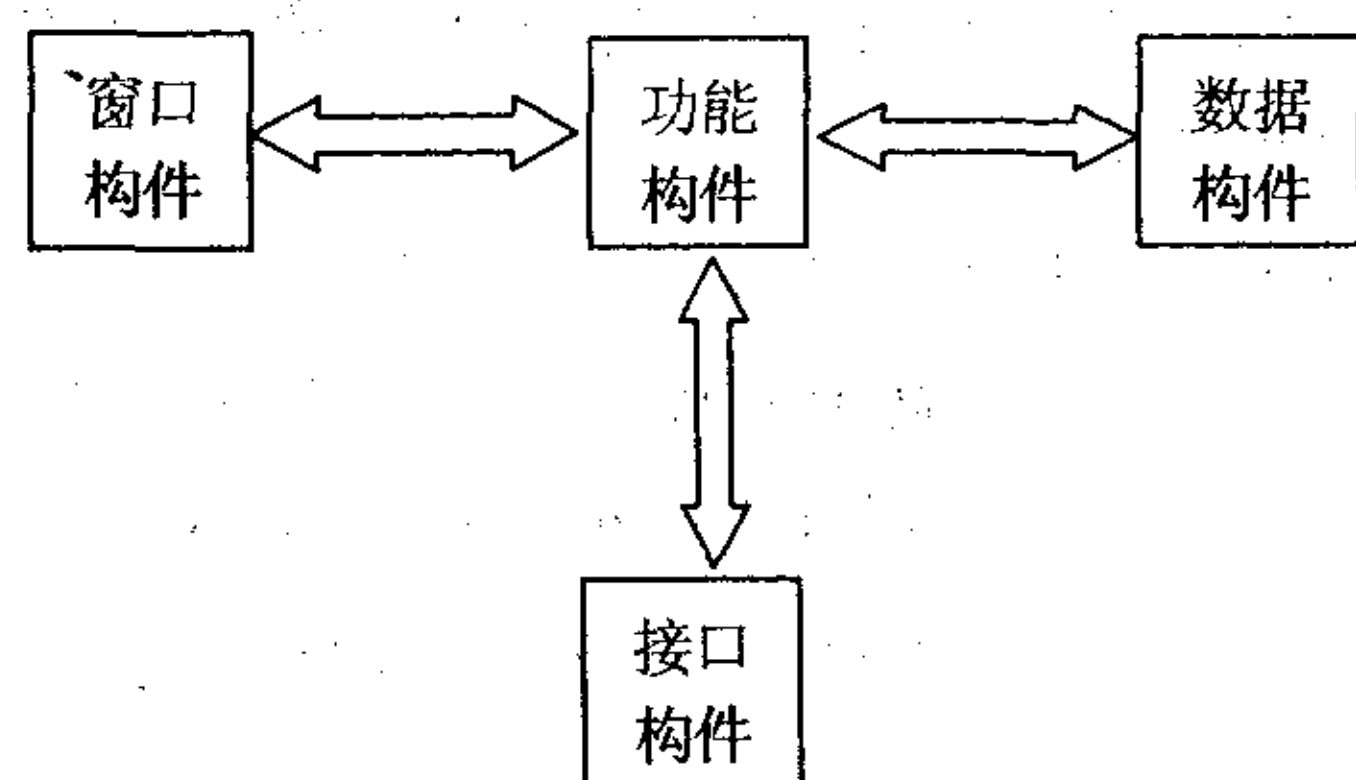


图1 计费系统中的构件

类似三层结构,它将系统可以分成三个独立的逻辑组件:学生组件、成绩查询组件和数据组件。它首先利于面向对象的模块化和数据封装,其次明显孤立的各个构件允许对一个构件的改动不影响其它构件,每个构件对于其它构件来说是一个黑箱,提供了一套接口,对于其它构件来说不需要了解该构件是如何完成该功能的。

其中数据构件由数据库设计提供,因此本次设计仅提供窗口构件及功能构件和数据接口构件。在进行系统设计时,窗口构件完成与用户界面的具体任务,使用户能够方便地操作,要求界面统一、简洁。由于PowerBuilder面向对象的特点主要体现在用户对象的技术运用上,故本系统在实现时窗口构件由窗口和一些可视用户对象组成,功能构件以及数据接口构件由不可视用户对象构成,在实现具体任务时窗口对象通过实例化功能对象和数据接口对象后调用功能对象与数据接口对象的方法从而实现对各种查询的处理。

### 3.2 学生成绩查询统计系统构件化实现

#### 3.2.1 PFC 简介

PFC(Powerbuilder Foundation Class, Powerbuilder 基础类库)<sup>[4]</sup>是PowerBuilder提供的一个类库,PowerBuilder把一些常用的、具有共性特征的对象(例如:窗口、菜单、数据窗口、用户对象等)提炼出来,并在每个对象中封装了不同的服务,构造了PFC基础类库。PFC基础类库完美地体现了面向对象程序设计的三方面重要特征,即继承性(Inheritance)、封装性(Encapsulation)和多态性(Polymorphism)。PFC是PowerBuilder面向对象技术的体现,它是一组典型的可复用构件。PFC的应用,实际上就是面向对象技术的应用。

PFC是一组典型的可复用构件,它提供了一些常用的函数和功能,有非常友好的用户界面,用PFC作为开发工具,可以节省很多的时间。在PFC基础类库中,各种基础类的对象均可以被继承,如窗口、菜单、结构、数据窗口、用户对象等等。由继承而来的子孙对象包含了祖先对象的所有属性、变量、函数、结构、控件和



程序,并可对它们进行修改和扩充。编写一个完全建构在 PFC 类库之上的应用可大大提高软件质量,而且利用基类的已有代码也会使得许多复杂的功能实现起来要轻松得多。

### 3.2.2 PFC 方法与传统方法的比较

当用 PFC 开发应用程序时,许多老的开发方法也都必须改变,这里概述了用 PFC 开发 PowerBuilder 应用程序时的一些主要的变化,同时对它们进行了简单的比较。

#### (1) 窗口上的对象。

当把一个对象放到窗口上时,应该使用的是 pfe-main.pbl 上的标准可视用户对象或它的一个子类,而不是在 pfcmain.pbl 中的对象。如果使用了一个在 pfcmain.pbl 中的对象,那么将切断其扩展层,并且使得以后对 pfemain.pbl 中对象的所有改变无效。

#### (2) 为菜单项的 Clicked 事件编程。

在以前,触发父窗口的保存事件的代码可能是:

```
ParentWindow.TriggerEvent("ue_save")
```

然而,在 PFC 中,有消息循环这样一个概念,这是一种在对象之间传递消息的方法,使用消息循环后,上面的代码将变成:

```
of_message("ue_save")
```

注意:这里没有指向父窗口。这是因为消息循环根据它自己的逻辑,把消息分配给不同的对象,直至可以接收消息的那个对象。

#### (3) 应用程序事件编写脚本。

我们过去常常在应用程序的 Open 事件中编写代码来实现从 INI 型文件读取 DBMS 连接参数,然后执行连接;在 System Error 事件中编写代码来显示一个错误;在 Close 事件中编写代码来做清理和中断数据库连接。当用 PFC 时就应该避免这样编写代码,应用程序事件脚本应该仅仅包含一个 CALL 语句,来调用 AppManager 对象的一个相应事件。

### 3.2.3 实例

构件化方法是通过用户对象实现的<sup>[5]</sup>,用户对象通过它的方法实现其所具有的一般的成绩查询功能(通过用户对象的函数实现),对每一个成绩查询功能可以分解为一个具体的处理动作,将这些部分构成单独的用户对象的方法,对于各种成绩查询的处理转化为对功能的处理。各个方法之间通过变量的修改来实现方法之间的通讯工作。用户对象通过它的服务实现与窗口对象有关的成绩查询处理(通过用户对象的事件实现)。

以数据接口构件中的标准可视用户对象为例,在学生成绩查询统计系统使用过程中,用户对所提取的

数据经常要进行批修改、查询、排序、删除等操作,这些操作都必须在数据窗口中完成,因此有必要对这些功能在数据窗口中进行封装。数据窗口对象由标准可视用户对象构成。对象的方法即完成数据窗口功能,通过增加多个 USER OBJECT FUNCTION 实现;数据窗口的属性由 INSTANCE 变量描述;数据窗口的服务由数据窗口对象的事件完成。

以下构造一个数据接口构件中的标准可视用户对象——Ue\_DataWindow。

#### (1) 属性:

应当能够描述数据窗口封装后为实现其功能所必须具有的特性,如为了实现条件提取,增加了如下属性:

①描述数据窗口内的表有几个字段:rowcolumnnumber;

②描述数据窗口每个字段的数据类型、字段名称等。

为了能够动态修改数据窗口语法,增加如下属性:描述数据窗口的 SQL 的语法:is\_oldsql

为得到窗口提取数据时是否被中断,增加如下属性:

描述数据窗口是否取消了提取数据:ib\_cancel。

#### (2) 服务:

RETRIEVESTART 设置 ib\_cancel 为 FALSE

RETRIEVEEND 设置 ib\_cancel 为 TRUE

CONSTRUCT 得到对数据窗口对象的具体描述(如有多少列,各个列的名称、数据类型等)

#### (3) 方法:

uf\_appendcondition( ) // 当数据窗口提取条件变化时将变化的条件附加到原数据窗口中

uf\_cancel( ) // 当提取数据时取消提取

uf\_getcolumnname( ) // 得到数据窗口的各个列名

uf\_getcolumnntype( ) // 得到数据窗口的各个列的数据类型

uf\_getdwcolumnname( ) // 得到数据窗口的各个字段的名称

uf\_getalltextname( ) // 得到文本编辑框内的文字内容

uf\_isbusy( ) // 得到系统是否忙

Uf\_enablemultiselect( BOOL) // 使数据窗口具有多行选择功能(选择时按 SHIFT 键。)当参数为 TRUE 时具有多行选择功能,否则不具有

Uf\_multiselect( ) // 多行选择功能的实现。

以上数据窗口对象的方法实现了数据窗口的大部分功能,每一个数据窗口可以由该用户对象继承,数据窗口的大部份功能由祖先实现,当发现该用户对象缺少某项功能时可在该用户对象上添加此功能,在子类

(下转第 220 页)



数据库支持如 SQL Server2000、Oracle8i、DB2 等。主要是存放系统数据和部分后台数据库逻辑程序。它与用户界面层和业务逻辑层分开,易于管理和维护,并且执行效率高、数据安全性好。

### 3.2 软件架构设计

基于上述的 B/S 下的系统结构,在设计软件系统架构时考虑以下几个方面来架构本系统,分别从纵向和横向两个方面分析,如图 5 和图 6 所示。

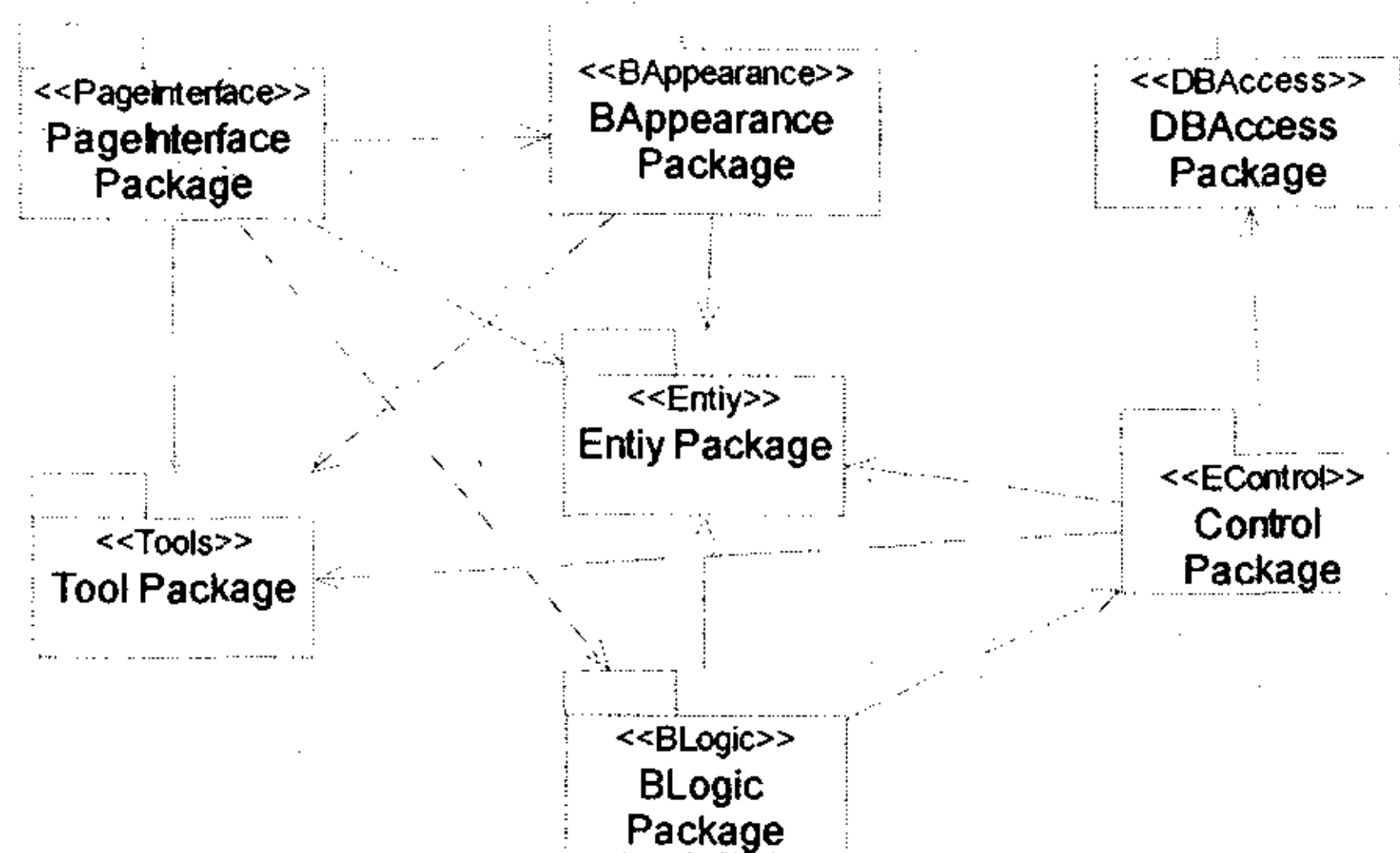


图 5 系统软件纵向架构图

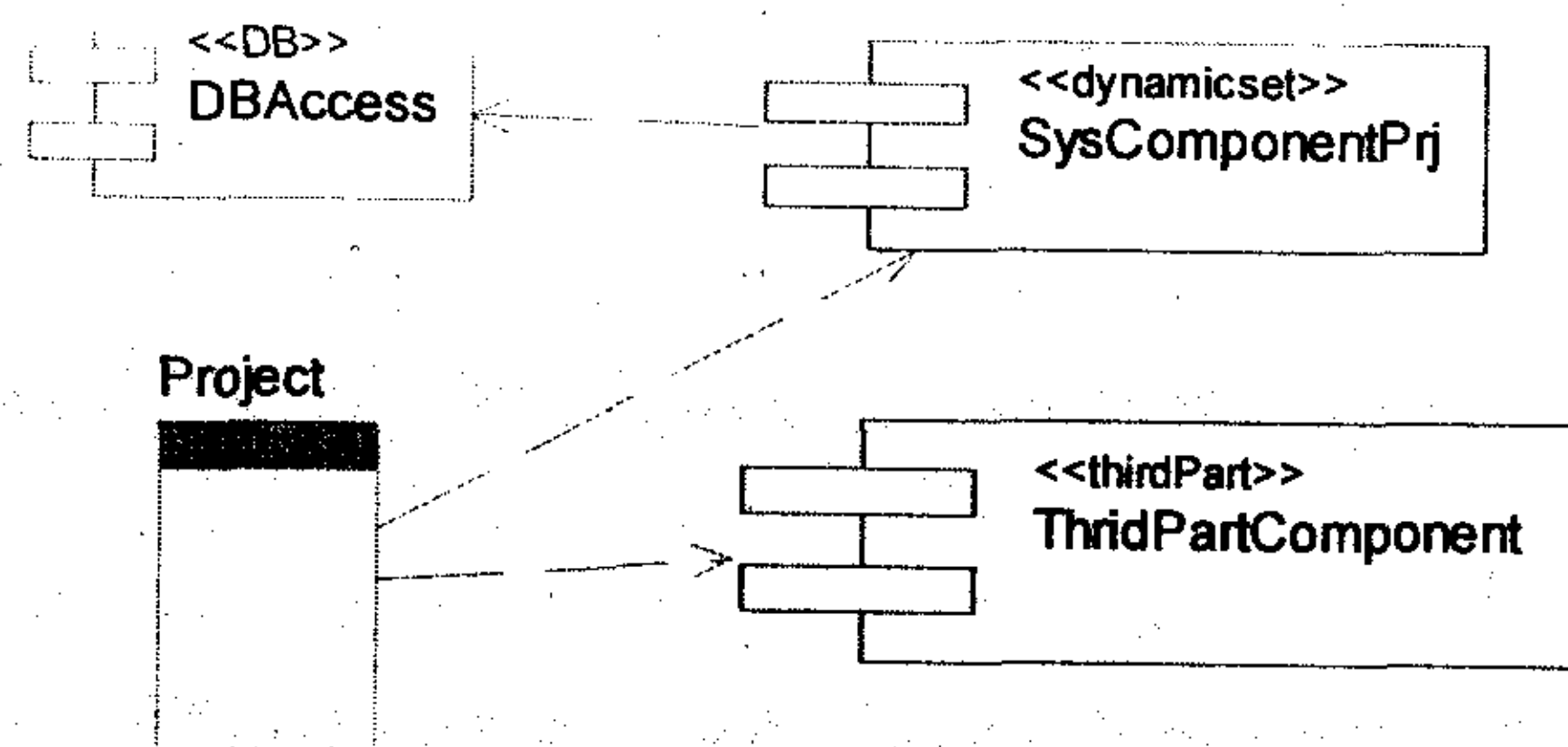


图 6 系统软件横向架构图

图 5 是从系统的纵向角度描述系统的架构,共有 7 个包,分别是页面接口包、实用工具包、业务外观包、业务逻辑包、实体包、实体控制包和数据访问包。package 和 package 之间具有依赖或引用关系。Tool

Package 和 Entity Package 引用较多,因为其他几个包都要使用它们。

图 6 是从系统的横向角度描述系统的架构,Project 表示系统运行项目,DBAccess 为数据库访问组件,SysComponentPrj 为系统组件群,ThirdPartComponent 为系统中所引用的第三方组件,各组件都是与 Project 级别相同,与 Project 同为一个解决方案下的不同项目,最终是以动态连接库的形式发布。

## 4 结束语

使用 UML 对 B/S 环境下的设备磨损判断和部位识别系统进行了分析和设计,给出了主要模型图和系统实现的结构模型,为后续的系统程式实现的过程起到较强的指导性作用,提高了系统的开发效率,缩短了开发周期。

### 参考文献:

- [1] Schmuller J. UML 基础、案例与应用[M]. 李 虎,王美英,万里威译.北京:人民邮电出版社,2000:5-10.
- [2] Gamma E, Ralph R H, Vlissides J J. Design Patterns Elements of Reusable Object-Oriented Software[M]. 李英军,马晓星译.北京:机械工业出版社,2000.
- [3] 蒋 慧,吴礼发,陈卫卫. UML Programming Guide 设计核心技术[M]. 北京:北京希望电子出版社,2001.
- [4] Larman C. UML 和模式应用-面向对象分析与设计导论[M]. 姚淑珍,李 虎等译.北京:机械工业出版社,2002.
- [5] Boggs W, Boggs M. UML 与 Rational Rose 2002 从入门到精通[M]. 邱仲潘 等译.北京:电子工业出版社,2002.
- [6] 程树林,毛 宁,蔡德辉,等. 基于 UML 的 .NET 系统建模方法[J]. 微电子学与计算机,2003(12):47-50.

(上接第 216 页)

中直接调用。如果父类对象中功能不完善,便可以直接修改父类对象的功能,以减少修改量。

## 4 结 语

软构件技术的出现,极大地满足了多个应用领域的要求,使得用各种技术形成的软构件可以最大程度地进行重用。这就引出了大规模软件开发所面临的另一个问题:如何建造面向对象的软构件库结构,并有效地组织和管理?在大型软构件库的支持下,利用现有的、质量好、可靠性强的软构件,按照大规模软件开发的工程规范进行开发,是满足这些大型系统要求的良好途径。现在人们越来越多地寄希望于具有分散和联合处理能力的开放性分布式软构件技术。分布式软构

件技术的目标是实现开放的软构件产品,使应用程序能相互操作,降低开发与管理费用。到目前为止,虽然还没有出现一个完整的开放式软构件系统的实施方案,但软构件化成为软件工业化将是必然的。

### 参考文献:

- [1] 杨美清. 软件复用技术[J]. 计算机科学,1999(5):21-26.
- [2] 梅 宏,常继传. 软件构件和软件构架技术[J]. 计算机世界,2000(8):30-34.
- [3] Pressman R S. 软件工程——实践者的研究方法[M]. 第 5 版. 梅宏译.北京:机械工业出版社,2002.
- [4] 何旭洪,余建英. PowerBuilder 8.0 数据库系统开发实例导航[M]. 北京:人民邮电出版社,2002.
- [5] Norman R J. Object-Oriented Systems Analysis and Design [M]. [s.l.]:Prentice-Hall International, Inc, 2000.